

Real-Time Vehicle Tracking on a Highway*

WEI-LIEH HSU, HSIAO-RONG TYAN**, YU-MING LIANG⁺

BOR-SHENN JENG⁺⁺ AND KUO-CHIN FAN

*Department of Computer Science and Information Engineering
National Central University*

Chungli, 320 Taiwan

^{**}*Department of Information and Computer Engineering*

Chung Yuan Christian University

Chungli, 320 Taiwan

⁺*Institute of Information Science*

Academia Sinica

Taipei, 115 Taiwan

⁺⁺*Telecommunication Laboratories*

Chunghwa Telecommunication Co., Ltd.

Taoyuan, 320 Taiwan

We propose a method which can be used to perform real-time tracking of moving vehicles on highways. In addition to tracking regular cars, the proposed method can also track a vehicle performing a lane change. The proposed method consists of two sections: a detection and a tracking. In the detection section, we use entropy-based features to check for the existence of vehicles. Then, we perform tracking based on the entropy features derived in the detection section. By conducting a great number of experiments, we have demonstrated the efficiency as well as the effectiveness of the proposed system.

Keyword: intelligent transportation system, entropy, real-time, vehicle tracking, surveillance system

1. INTRODUCTION

In recent years, automated traffic surveillance conducted using electronic devices has become an important area of development in the Intelligent Transportation System field. In different types of ITS-related applications, the tracking of moving vehicles plays an important role. By tracking a vehicle, one is not only able to detect some illegal events, such as speeding or illegal lane change, but also calculate a number of traffic parameters, such as the traffic flow or the average vehicle speed. Since the development of an ITS-based city will be an important step in urban modernization, many research institutes and private organizations have devoted great efforts to exploring related problems over the past decade [1-5, 9, 10, 12, 13]. In [1], Beymer *et al.* proposed a feature-based method for detecting and tracking moving vehicles in real time. Instead of tracking an entire vehicle, they chose to track some features that were less sensitive to the problem of partial occlusion. Their method works during both daytime and at night. However, the

Received March 29, 2004; revised April 20, 2004; accepted May 3, 2004.

Communicated by H. Y. Mark Liao.

* This paper has appeared in the Proceedings of IEEE 6th International Conference on Intelligent Transportation System, 2003, pp. 909-914.

time complexity of their method is high. In [3], Masound *et al.* proposed a vision-based system for monitoring traffic in weaving road sections. The system consists of three levels: a feature level, blob level and vehicle level. They use a simple rectangular patch together with a dynamic behavior model to characterize a moving vehicle. Then, the tracking process is accomplished by applying an extended Kalman filter. Sullivan *et al.* [4] developed a simplified version of a 3D model-based approach to track a vehicle. The drawbacks of their approach are that a large set of models is required and the computation involved is intensive. In [5], Tseng *et al.* applied the snake model to extract the contours of vehicles, and then used these contours as the bases for tracking vehicles. However, these features are very sensitive to occlusion, noise and ambient lighting conditions.

In this paper, we propose a new vision-based system that can perform real-time vehicle detection and tracking. In the detection phase, we use entropy-based features to check for the existence of vehicles [12, 13]. Then, we perform tracking by using the above mentioned entropy-related features in combination with the leading edges detected from target vehicles. From the experimental results, we have demonstrated the efficiency as well as the effectiveness of the proposed system. The rest of this paper is organized as follows. In section 2, we provide a brief introduction to the underlying theory that is the basis of the proposed system. The proposed method is presented in section 3. Finally, experimental results and conclusions are given in sections 4 and 5, respectively.

2. UNDERLYING THEORY

In this section, the entropy theory that is used to define a feature for checking for the existence of vehicles and performing real time tracking are introduced. Entropy is a macro parameter that is commonly adopted to describe the degree of disorder of a system. In this paper, we use the concept of entropy to judge the existence of a vehicle. In thermodynamics, the entropy of a system is usually defined as [11]

$$S = k \ln \Omega, \quad (1)$$

where Ω is the probability of occurrence of a message. k and S represent a constant and the entropy respectively. When the internal energy of a system increases, its entropy increases as well. Usually, equilibrium in a system will be reached when the constituent molecules within the system are uniformly distributed. Under these circumstances, the entropy reaches its maximum. Shannon's entropy, which is commonly used in information theory, is based on the concept that the gain in information from an event is inversely proportional to the probability of occurrence of a message, p_i . Therefore, Shannon's entropy can be defined as [8]

$$H = - \sum_{i=1}^n p_i \log p_i. \quad (2)$$

It is assumed here that the probability distribution $\sum_{i=1}^n p_i = 1$. Under this assumption, the probability of every component must be greater than zero. Since the corresponding in-

formation gain cannot be defined when a probability goes to zero, we adopt the exponential entropy defined by Pal and Pal [8] as our entropy measurement tool. In their approach, the information gain is defined as

$$I = \exp(1 - p_i). \quad (3)$$

Furthermore, the information gain that can be derived from a message is inversely proportional to its probability of occurrence, p_i . Therefore, the exponential entropy can be defined as the expected value of the gain function, i.e.,

$$H = E(I) = \sum_{i=1}^n p_i \exp(1 - p_i). \quad (4)$$

In thermodynamics [11], if a system absorbs a small amount of heat (energy), dQ , then the entropy change of this system is

$$dH = \frac{dQ}{T}, \quad (5)$$

where T is temperature. It is apparent that the entropy change of a system is related to its energy change. In this paper, a moving vehicle running on a lane is considered to be the transferring energy; thus, we use the entropy to check for its existence and, in the meanwhile, track its trajectory. Entropy has a property worth noting [12], which is based on the concept that if the detection zone contains more blocks having connected components, then its corresponding entropy is larger. Using this property, we are able to quickly check for the existence of a vehicle.

3. THE PROPOSED METHOD

In a typical traffic surveillance system, a camera is mounted high above and beside a road. In order to extract useful traffic parameters, in the first step, we use interframe differencing to derive the difference between the current and the previous frames, and we then binarize it to obtain a foreground object map as illustrated in Fig. 1. The equation for the subtracting process is

$$M(p) = \begin{cases} 1 & \text{if } |I_k(p) - I_{k-1}(p)| > I_{th}; \\ 0 & \text{if } |I_k(p) - I_{k-1}(p)| \leq I_{th}, \end{cases} \quad (6)$$

where $I_k(p)$ and $I_{k-1}(p)$ indicate the intensity measured at pixel p of frames k and $k - 1$, respectively. M here represents a binary image, and I_{th} represents a pre-determined threshold.

We divide the subsequent tracking process into two steps: vehicle detection and vehicle tracking. The details of these two steps are given in the following section.

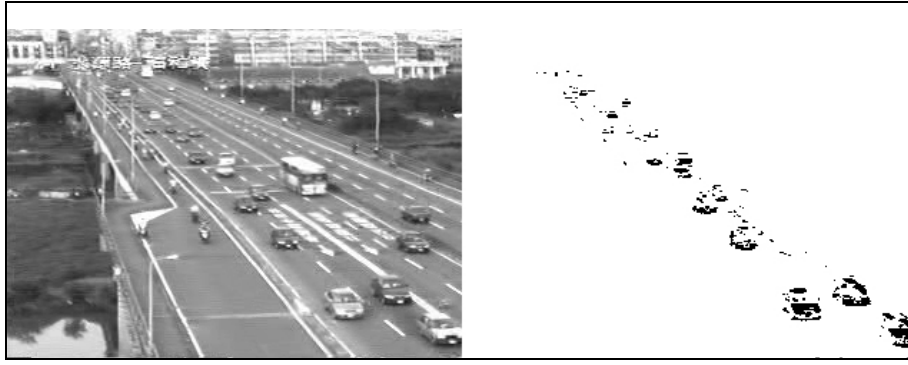


Fig. 1. Left: the original image. Right: the corresponding foreground object map.

3.1 Vehicle Detection

In order to correctly detect a moving vehicle, we preset two detection zones in the same lane and set the distance between them to D_i (which will be explained in detail later). A vehicle detection zone can be divided into n blocks as shown in Fig. 2, and the probability P_i of each block is

$$P_i = \frac{N_i}{N}, \quad (7)$$

where $N = \sum_{i=1}^n N_i$,

N_i is the number of active pixels in the i^{th} block of the detection zone, and N indicates the total number of active pixels in the detection zone. Under these circumstances, the overall probability and the exponential entropy of the detection zone can be easily computed using Eq. (4).

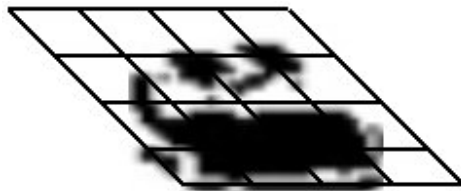


Fig. 2. Active pixel distribution in a detection zone.

When a vehicle passes through a detection zone, the variation of its corresponding exponential entropy value with respect to the frame number is that shown in Fig. 3. In order to check for the existence of a vehicle, we can check the variation of the exponential entropy within one of the detection zones over a period of time. The detection zone is

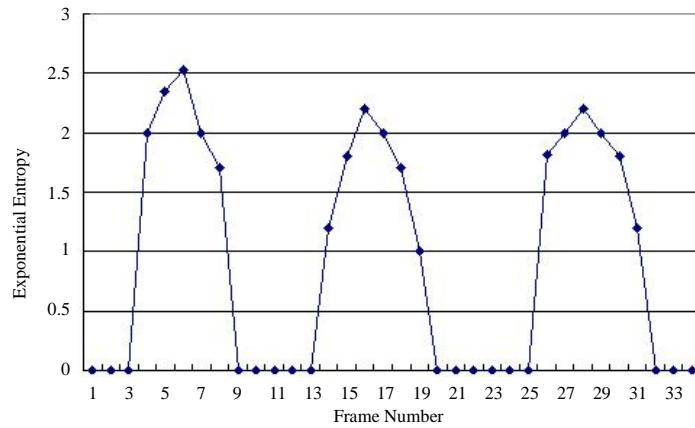


Fig. 3. The variation of the exponential entropy occurring within a short time duration.

partitioned into n blocks. When there are no vehicles, the corresponding exponential entropy will be zero. However, if a vehicle moves into the detection zone and starts to occupy one block, then the exponential entropy jumps from 0 to 1 immediately based on Eq. (4). When the exponential entropy reaches or comes very close to the maximum value, i.e., 2.718(e), this indicates that the detection zone is occupied by a vehicle. The exponential entropy value drops to 0 once the vehicle leaves the detection zone. The above behavior can be characterized by using a mathematical model. The modeling process is not a trivial task because the curve of exponential entropy has two discontinuities. These discontinuities correspond to the exponential entropies detected at the instants when a vehicle is entering and leaving a detection zone. Let f be a variable that represents the frame number and $H(f)$ be the exponential entropy at the instant of frame f . Assume that it takes $2T$ frames for a vehicle to pass through a detection zone. If the vehicle moves at a constant speed, then after a period of T frames, the exponential entropy will reach or come very close to the maximum value, $H_m = 2.718$. Let the frame corresponding to the highest entropy value be f_h . The entropy for a car entering or leading a detection zone is characterized by

$$H(f) = \begin{cases} G(f) & \text{if } G(f) \geq 1; \\ 0 & \text{if } G(f) < 1, \end{cases} \tag{8}$$

where $G(f) = H_m \exp\left(-\left(\frac{f - f_h}{T}\right)^2\right)$.

The physical meaning of Eq. (8) can be analyzed with the help of Fig. 4. In fact, the value of $H(f)$ is regulated by $G(f)$. If f is outside the $2T$ time duration (i.e., not in the detection zone), then $G(f)$ will be less than 1, and $H(f)$ will be 0. At frame $f_h - T$, a vehicle is just entering the detection zone, and the exponential entropy is 1. If a vehicle is detected at an instant between frames $f_h - T$ and f_h , then its exponential entropy will be larger than 1 but less than 2.718.

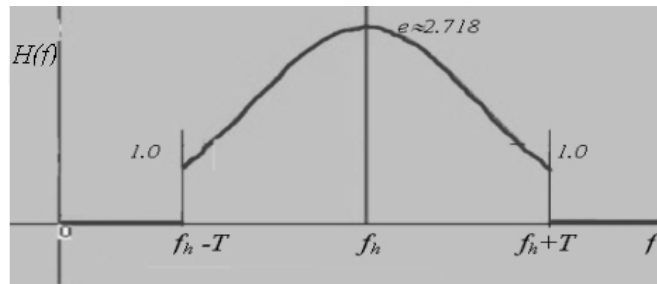


Fig. 4. Distribution of $H(f)$. The horizontal axis indicates the frame number, and the vertical axis represents the exponential entropy value.

According to the previous discussion, we know that when a vehicle passes through a preset detection zone, its corresponding frame by frame exponential entropy will change from 1 to at most 2.718 and then back to 1. When a vehicle leaves the detection zone, the exponential entropy will soon drop to zero. To effectively detect vehicles in the implementation, we also make use of a special feature which is tightly associated with a vehicle, the detected front edge caused by the shadow under the moving vehicle. This feature is very stable and is suitable under most weather conditions. Some detected leading edges of a number of moving vehicles are shown in Fig. 5.



Fig. 5. Detected leading edges of moving vehicles.

In order to correctly compute the velocity of a vehicle, we calculate the time duration, T_d , required by a vehicle to pass through two preset detection zones. When the leading edge of a vehicle is aligned with the front margin of the first detection zone (see Fig. 6), we register the corresponding frame number. When the front edge of the same vehicle moves forward and aligns with the front margin of the second detection zone, we record the frame number again. T_d can be calculated using the difference between these two frame numbers and then used to calculate the speed of a vehicle. If the distance between the two detection zones is D_i , then the speed, v , of a vehicle that passes through the two detection zones can be computed by

$$v = \frac{D_i}{T_d}. \quad (9)$$

In our implementation, if a vehicle is speeding, it will trigger a tracking process. The details of the tracking phase are given in the next section.

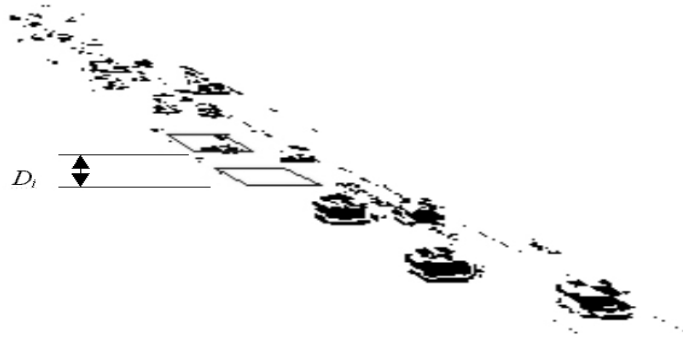


Fig. 6. Two detection zones and their distance D_i .

3.2 Vehicle Tracking

The Kalman filter is usually used to track a moving object. The basic operation of a Kalman filter depends on trying to minimize mean square difference between measured features and predicted features by adjusting filter parameters. If a Kalman filter is used to track a fast moving vehicle with distance feature, it may fail and track the wrong vehicle. If more features are added, for example, color or shape, the performance will improve. However, the computation time will rise as well. We use the velocity of a vehicle to predict its location at the next frame. Since vehicles usually have relatively small changes in speed between two consecutive frames, the prediction results are very accurate in most cases.

In order to perform accurate vehicle tracking, the effects of perspective projection must be considered. We use a bounding box to track a target vehicle. When a tracked vehicle is moving toward the viewer, its bounding box should be adjusted based on the perspective projection rule. In addition, the concentration of active pixels enclosed within the bounding box is used as an indicator to guide the tracking process.

Fig. 7 shows a number of vehicles moving from the northwest corner (coordinate (0, 0)) toward the southeast corner. The locations of the bounding box in two consecutive frames are illustrated in Fig. 8. Basically, the moving distance appears longer and longer as the vehicle travels toward the southeast corner. This phenomenon is caused by the perspective projection.

Since some visual features, such as color or shape, cannot be used reliably to perform tracking, we use the vehicle's velocity instead to estimate the distance moved by the bounding box between two contiguous frames. We adopt the pinhole camera model to compensate for perspective projection. After the size of the box is adjusted frame by



Fig. 7. Vehicles moving from the northwest corner to the southeast corner.

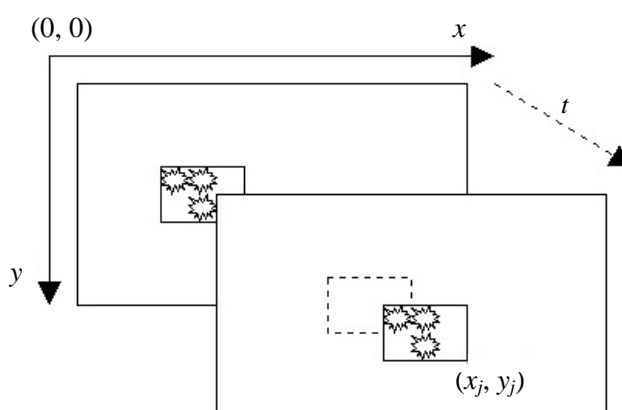


Fig. 8. Movement of the bounding box with respect to the time and spatial axis.

frame, the next step is to move the bounding box to its accurate horizontal position with the help of entropy and the front edge of the bounding box. We shall describe this step in detail later.

For vehicle tracking, we track along the x - and y -axis directions. First, we transform the world coordinates into the camera coordinates. Fig. 9 illustrates the camera setup and the relationship between the world coordinate system and the camera coordinate system.

Since with a high frame rate the time interval between two contiguous frames is short, it is reasonable to assume that the vehicle is moving at a constant velocity, $\vec{v}^w(t)$, along a certain direction in the world coordinate system. Hence, $\vec{v}^w(t+1) = \vec{v}^w(t)$. For simplicity, we align the image plane of the camera coordinate system $(\hat{c}_x, \hat{c}_y, \hat{c}_z)$ with the xy plane, and align the optical axis with the z -axis. Since any motion can be decomposed into three orthogonal directions, we assume that the rectangular box that is parallel to the image plane has a velocity $\vec{v}^c(t)$. In addition, its z component, $\vec{v}_z^c(t)$, is perpendicular to the image plane. The x and y components, $\vec{v}_x^c(t)$ and $\vec{v}_y^c(t)$, are in the image plane. The world coordinates and the camera coordinates basically have no relative motion.

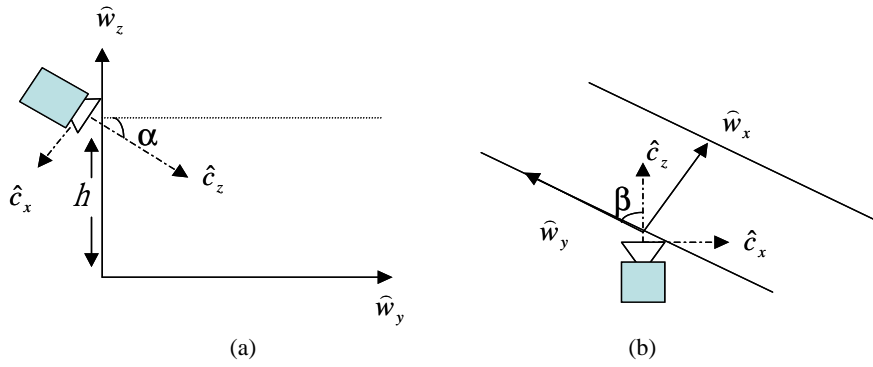


Fig. 9. (a) Side-view of camera setup and world coordinates; (b) Top-view of camera setup and world coordinates. β and α are the pan angle and the tilt angle respectively; h is the height of camera.

Therefore, when we transform the world coordinates into the camera coordinates, the value of $\vec{v}^w(t+1)$ will not be equal to $\vec{v}^c(t+1)$, but rather $\vec{v}^c(t+1) = \vec{v}^c(t)$. The above explanation indicates that the distance moved by a bounding box remains constant in the camera coordinate system between any two contiguous frames, that is, $d^c(t+1) = d^c(t)$.

To derive the z -component of the rectangular box in the camera coordinate system, we project the camera coordinates onto the image plane and find the corresponding z -component, z^c . The mapping step that brings things onto the image coordinates is accomplished by using a pinhole camera model. Thus, the distance moved by the bounding box along the x - and y -directions between two contiguous frames can be estimated as

$$\begin{aligned} dx_t^i &\approx f_c \frac{dx^c}{z^c}, \\ dy_t^i &\approx f_c \frac{dy^c}{z^c}, \end{aligned} \quad (10)$$

where dx^c and dy^c represent the distance moved by the bounding box between two consecutive frames along the x - and y -axes in the camera coordinate, respectively. Similarly, dx_t^i and dy_t^i are the distance moved by the rectangular box along the x - and y -axes in image coordinates over time interval t to $t+1$, and f_c is the focal length of the camera. From Eq. (10), it is obvious that the bounding box depends on the value of z^c . While the value of z^c cannot be measured directly from an image frame, it can be estimated from the width of a road lane.

After we transform the world coordinates into camera coordinates, the width of a road, Wid_z^c , is kept invariant. We consider the case that the optical axis of the camera lens is not parallel to the road, so we assume the pan angle of a camera is β' in the camera coordinate system. Then the parallel component of the lane width is $Wid_z^c / \cos \beta'$ as illustrated in Fig. 10.

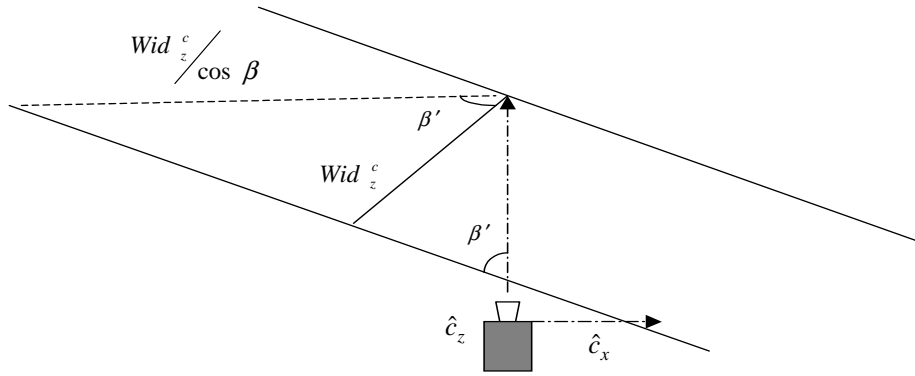


Fig. 10. Relationship between the optical axis of the camera's lens and lane in the camera coordinate.

At time t , the bounding box moves toward the image plane along the z^c direction axis. Based on the pinhole camera model, the parallel component of the lane width located in the image plane, Wid_t^i can be computed by

$$Wid_t^i = f_c \frac{Wid_z^c / \cos \beta'}{z^c}. \quad (11)$$

Substituting Eq. (11) into Eq. (10), we get

$$\begin{aligned} dx_t^i &= \frac{\cos \beta' \cdot dx^c}{Wid^c} Wid_t^i, \\ dy_t^i &= \frac{\cos \beta' \cdot dy^c}{Wid^c} Wid_t^i. \end{aligned} \quad (12)$$

For computing the lane width, Wid_t^i , we can compute the width between the right and the left boundaries of a lane as illustrated in Fig. 11. These boundaries can be determined based on the slope of the lane boundary and its y intercept.

From Fig. 11, it is obvious that the width of a lane depends on the location of a moving car. According to Eq. (12), the moving distance of the bounding box along x -axis and y -axis is

$$\begin{aligned} \frac{dx_t^i}{dx_{t-1}^i} &= \frac{Wid_t^i}{Wid_{t-1}^i}, \\ \frac{dy_t^i}{dy_{t-1}^i} &= \frac{Wid_t^i}{Wid_{t-1}^i}. \end{aligned} \quad (13)$$

Rewriting Eq. (13), we have



Fig. 11. Vehicles move from northwest to southeast; dashed lines indicate the width of a lane at different locations.

$$\begin{aligned}
 dx_t^i &= \frac{Wid_t^i}{Wid_{t-1}^i} \cdot dx_{t-1}^i, \\
 dy_t^i &= \frac{Wid_t^i}{Wid_{t-1}^i} \cdot dy_{t-1}^i,
 \end{aligned}
 \tag{14}$$

where dx_{t-1}^i and dx_t^i represent the distances moved by the rectangular box along the x -axis between the previous frame interval and the current frame interval. Similarly, dy_{t-1}^i and dy_t^i represent the distances moved by the rectangular box along the y -axis between the previous frame interval and the current frame interval. Wid_{t-1}^i and Wid_t^i represent the lane widths at the times $t - 1$ and t , respectively.

When the vehicle tracking process starts, we set time $t = 0$, and let Wid_{-1}^i and Wid_0^i represent the width of the road lane at the rear and front margins of the second detection zone, respectively. dy_{-1}^i represents the distance moved between two contiguous image frames along the y -direction. Since different vehicles may have different velocities, the number of frames need to capture the behavior of a vehicle moving in the x - and y - directions may vary. Let the number of frames used to describe the speed of some vehicle moving in the y -direction be N . Then, $dy_{-1}^i = \frac{P_D}{N}$, where P_D is the total number of active pixels found between the two detection zones along the y -direction. Having dy_{-1}^i , we are able to derive dx_{-1}^i based on the relation shown in Fig. 12,

$$dx_{-1}^i = \frac{dy_{-1}^i}{\tan \theta}.
 \tag{15}$$

Using the above equation, we are able to estimate the distance moved by a bounding box between any two image frames correctly and efficiently. The tracking process is

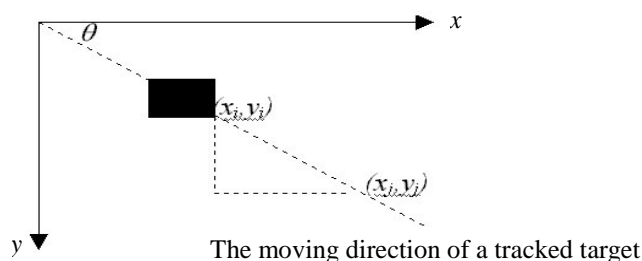


Fig. 12. The relationship among the direction of motion of a tracked target, the x -axis and y -axis.

divided into two steps. In the first step, we predict the location of the bounding box in the y -direction. We have mentioned in the previous section that the detected leading edge of a moving vehicle can be used to track a vehicle. Therefore, in the process of predicting the y -coordinate of a bounding box that encloses a moving target vehicle, we try to align the front side of the box with the leading edge of a tracked vehicle. In the case of Fig. 13 (a), the bounding box will be shifted down to make its lower side align with the leading edge of tracked vehicle. On the other hand, if the situation is like the case shown in Fig. 13 (b), the box will be shifted up. All the adjustments mentioned above are based on the value of entropy and the alignment between the box and the front edge of a vehicle.



Fig. 13. The relationship between the bounding box and a target vehicle (a) box moves too little. (b) box moves to far.

After the y -coordinate of the bounding box is determined, the next step is to calculate its x -coordinate. Since a tracked vehicle may change lanes during the tracking period, we have to adjust the horizontal location of the bounding box. Here, we use the entropy property. As explained in the previous section, if there are many blocks in the detection zone containing connected components, then its corresponding exponential entropy will be large. Based on this criterion, we are able to shift a bounding box horizontally and finally decide on the best matched bounding box. Fig. 14 shows the exponential entropy corresponding to the cases when the bounding box is located at three different horizontal positions. From Figs. 14 (a)-(c), it is obvious that when the target vehicle is better aligned with its bounding box, the exponential entropy is larger.

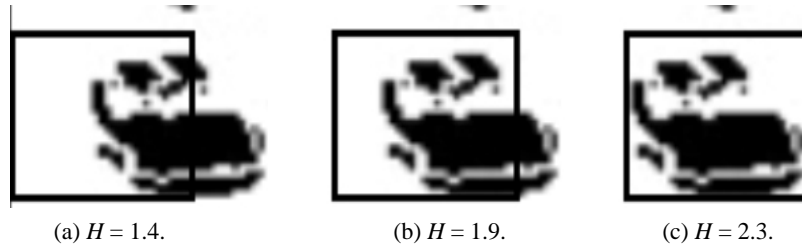


Fig. 14. The better the alignment between the vehicle and its bounding box, the higher the exponential entropy.

3.3 Determining the Size of a Bounding Box

As mentioned in the previous section, the size of a bounding box is subject to change due to perspective projection. Since the size of a bounding box will affect the calculation of the exponential entropy, it has to be updated frame by frame. Here we shall provide a way to calculate the size of a bounding box at any position. Actually, the size of a vehicle at any position is related to the perspective relationship between the real world space and the image plane. Thus, we assume that a rectangular box is parallel to the image plane at coordinate z^c , with width W^c and length L^c . When a rectangular box is projected onto the image plane, the widths of a lane at different times, $t - 1$ and t , can be calculated

$$W_{t-1}^i = f_c \frac{W^c}{z_{t-1}^c}, \quad W_t^i = f_c \frac{W^c}{z_t^c}. \tag{16}$$

Based on Eq. (11) the widths of a lane at distinct locations are inversely proportional to z^c . Therefore, by substituting Eq. (16) into Eq. (10), we obtain

$$\frac{W_{t-1}^i}{W_t^i} = \frac{Wid_{t-1}^i}{Wid_t^i}. \tag{17}$$

In order to keep the same length-width ratio of a tracked box, the following relation must be satisfied

$$\frac{L_{t-1}^i}{W_{t-1}^i} = \frac{L_t^i}{W_t^i}. \tag{18}$$

Therefore, we have

$$\frac{L_{t-1}^i}{L_t^i} = \frac{Wid_{t-1}^i}{Wid_t^i}. \tag{19}$$

The above equation indicates that the length and width of a bounding box are proportional to the width of a road lane in an image. When the tracking process is triggered, the length L_{t-1} and width W_{t-1} of the bounding box can be derived. Meanwhile, the width of a road lane, Wid_{t-1} , can be measured from an image frame. With the above information, we are able to derive the width of a road lane at any location. Meanwhile, we can update the size of a bounding box frame by frame and then compute its corresponding exponential entropy.

4. EXPERIMENTAL RESULTS

In the experiment we adopted several real video sequences that were provided by the transportation bureau of the Taipei City Government to test our algorithm. These video sequences were taken by a camcorder mounted above and beside the road. The frame rate was 10 frames/sec, and the width and height of each frame were 320 and 240 pixels, respectively. Our system was able to process each frame at a rate of 0.11 second/frame. Figs. 15 (a)-(f) is a typical example showing how our tracking algorithm works. After a vehicle moved through the two detection zones (Fig. 15 (a)), it (in the bounding box) was identified as a candidate car and then tracked. In the subsequent frames (Figs. 15 (b)-(f)), the tracked vehicle changed lanes, and our tracking algorithm was able to correctly track it.



Fig. 15. Our tracking algorithm is able to track a vehicle that performs a lane change.



Fig. 15. (Cont'd) Our tracking algorithm is able to track a vehicle that performs a lane change.

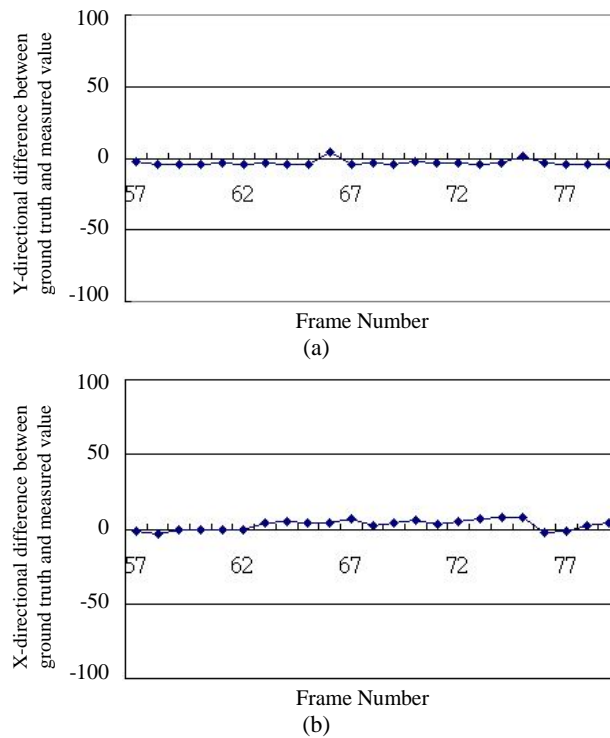


Fig. 16. The horizontal axis represents the frame number. (a) The vertical axis indicates the difference in pixel between the front margin of a bounding box and the front edge of a moving box. (b) The vertical axis represents the difference in pixels between the right lateral margin of the bounding box and the position of the most right pixel included in a target vehicle.

In order to verify the correctness of our tracking algorithm, we conducted another experiment. In the experiment, we checked to see if the front edge of a tracked vehicle was able to align with the front margin of a bounding box that enclosed the vehicle. Fig. 16 and Table 1 show the results obtained in this experiment. The horizontal axis in

Table 1. The x - and y - directional distances from the point of origin for the margin of a bounding box and the front edge of a moving box and their differences.

Frame Number	x -directional position of right lateral pixels of target vehicle	x -directional position of most right margin of bounding box	Difference of x - direction	y -directional distance of front edge of target vehicle	y -directional distance of front margin of bounding box	Difference of y - direction
57	176	177	- 1	129	131	- 2
58	178	181	- 3	131	135	- 4
59	184	184	0	134	138	- 4
60	187	187	0	137	141	- 4
61	187	187	0	138	141	- 3
62	190	190	0	140	144	- 4
63	196	192	4	143	146	- 3
64	201	196	5	146	150	- 4
65	204	200	4	150	154	- 4
66	210	206	4	154	150	4
67	218	211	7	157	161	- 4
68	221	219	2	162	165	- 3
69	227	223	4	165	169	- 4
70	233	227	6	171	173	- 2
71	239	236	3	174	177	- 3
72	247	242	5	180	183	- 3
73	254	247	7	184	188	- 4
74	259	251	8	189	192	- 3
75	267	259	8	196	195	1
76	274	276	- 2	200	203	- 3
77	283	284	- 1	207	211	- 4
78	292	290	2	213	217	- 4
79	301	297	4	220	224	- 4

Fig. 16 (a) represents the frame number, and the vertical axis indicates the difference (in pixels) between the front margin of a bounding box and the distance of the front edge of a moving box along y -direction. The vertical axis in Fig. 16 (b) represents the difference (in pixels) between the right lateral margin of bounding box and the position of the right most pixel covered by a target vehicle. It is obvious that these differences are very small. From this result, it is apparent that the proposed tracking algorithm is effective.

We also conducted a set of experiments to verify that our tracking approach was able to handle noisy situations. Fig. 17 shows a sequence of binarized images. When images were taken, there was a lot of jitter. Figs. 17 (a) and (d) show two frames that had a lot of noise caused by significant image jitter. The experimental results show that the proposed tracking algorithm was still successful under such bad situations.

In order to check whether the entropy-based approach is suitable for different lighting conditions, we also conducted a great number of experiments using different threshold values of interframe differencing, I_{th} (as defined in Eq. (6)). Fig. 18 shows the

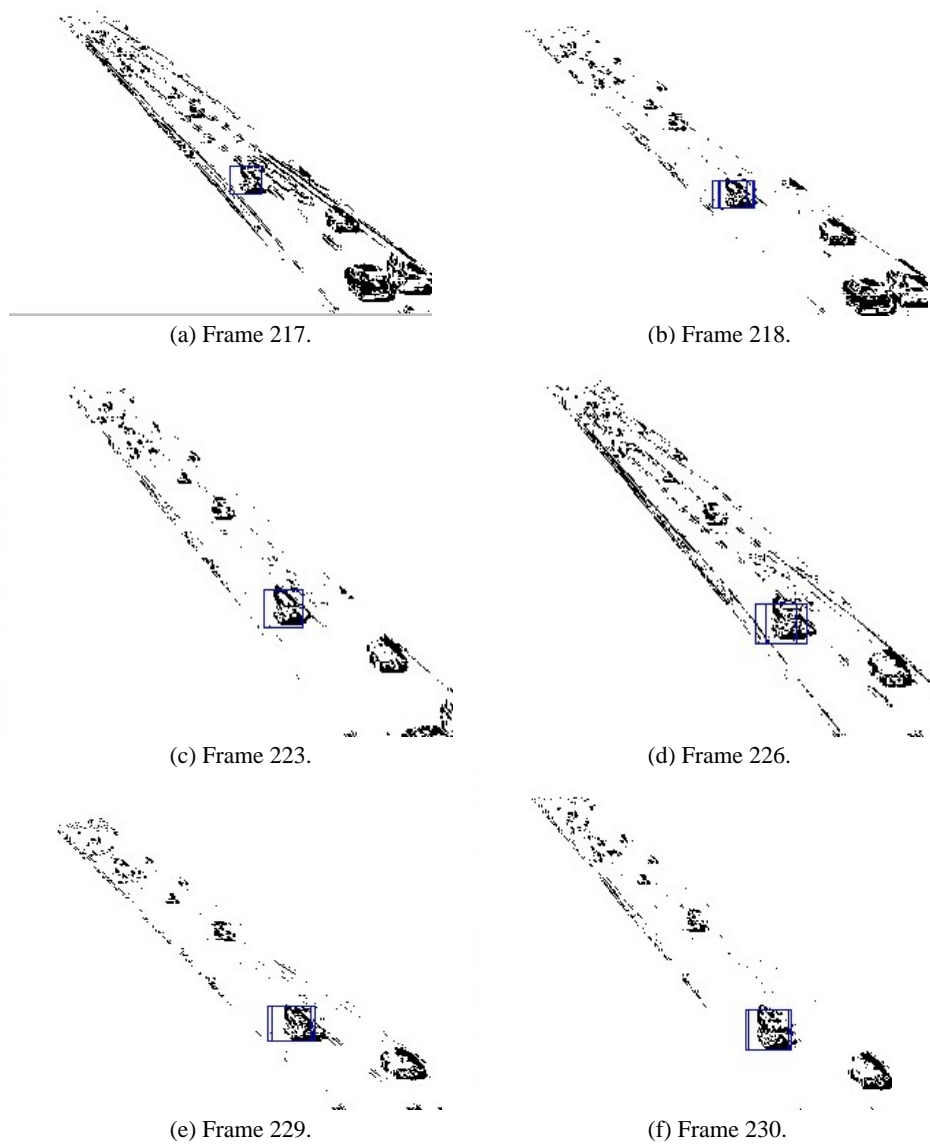


Fig. 17. Tracking results obtained under noisy conditions. In (b), (d) and (e), the overlapping boxes indicate that our algorithm can adjust the tracking position.

success rates achieved when different threshold values were used. A frame was considered successful if its corresponding bounding box covered 95% area of a tracked target and the bottom margin of this box aligned with the front edge of the target. From this set of experiments, it is obvious that the range of allowable I_{th} values is broad (from 20 to 80). With this broad allowable range, we can always set I_{th} to 50 so that the system will be as stable as possible. In addition, the broad allowable I_{th} range implies that the selected entropy-based features are insensitive to varying lighting conditions.

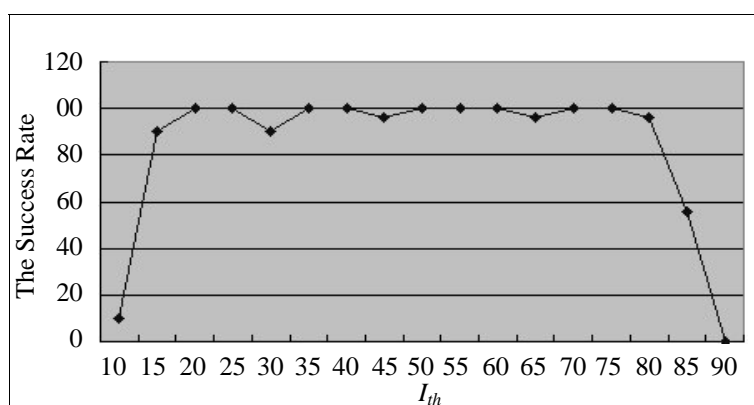


Fig. 18. Accuracy rate achieved when different threshold values were used for interframe differencing.

5. CONCLUSIONS

We have proposed a method which can be used to perform real-time vehicle detection and tracking. In addition to detecting vehicles, the proposed approach is also able to track a vehicle performing a lane change. The proposed approach consists of two phases: a detection phase and a tracking phase. In the detection phase of the proposed method, we use entropy to check for the existence of a vehicle. In the tracking phase, the entropy-based features are used to track the moving vehicles. Through a large number of experiments conducted using real data, we have demonstrated that the proposed method is efficient and accurate.

REFERENCES

1. B. Coifman, D. Beymer, P. McLauchlan, and J. Malik, "A real-time computer vision system for vehicle tracking and traffic surveillance," *Transportation Research Record Part C*, Vol. 6, 1998, pp. 271-288.
2. D. J. Dailey, F. W. Cathey, and S. Pumrin, "An algorithm to estimate mean traffic speed using uncalibrated cameras," *IEEE Transactions on Intelligent Transportation System*, Vol. 1, 2000, pp. 89-107.
3. O. Masound, N. P. Papanikolopoulos, and E. Kwon, "The use of computer vision in monitoring weaving section," *IEEE Transactions on Intelligent Transportation System*, Vol. 2, 2001, pp. 18-25.
4. G. D. Sullivan, K. D. Baker, A. D. Worrall, C. I. Attwood, and P. M. Remagnino, "Model-based vehicle detection and classification using orthographic approximations," *Image and Vision Computing*, Vol. 15, 1997, pp. 649-654.
5. S. T. Tseng and K. T. Song, "Real-time image tracking for traffic monitoring," in *Proceedings of IEEE 5th International Conference on Intelligent Transportation Systems*, 2002, pp. 10-14.
6. M. Ferraro, G. Boccignone, and T. Caelli, "On the representation of image structures via scale space entropy conditions," *IEEE Transactions on Pattern Analysis and*

- Machine Intelligence*, Vol. 21, 1999, pp. 1199-1203.
7. Y. Iwasaki, "A measurement method of pedestrian traffic flows by use of image processing and its application to a pedestrian traffic signal control," in *Proceedings of 2nd IEEE Conference on Intelligent Transportation Systems*, 1999, pp. 310-313.
 8. N. R. Pal, "Entropy: a new definition and its applications," *IEEE Transactions on System, Man and Cybernetics*, Vol. 21, 1993, pp. 1260-1270.
 9. H. Mori, M. Charkari, and T. Matsushita, "On-line vehicle and pedestrian detections based on sign pattern," *IEEE Transactions on Industrial Electronics*, Vol. 41, 1994, pp. 384-391.
 10. E. K. Bas and J. D. Crisman, "An easy to install camera calibration for traffic monitoring," in *Proceedings of IEEE International Conference on Intelligent Transportation Systems*, 1997, pp. 362-366.
 11. D. V. Schroeder, *An Introduction to Thermal Physics*, Addison Wesley, San Francisco, 2000.
 12. W. L. Hsu, H. Y. Mark Liao, B. S. Jeng, and K. C. Fan, "Real-time traffic parameter extraction using entropy," *IEE Proceedings – Vision, Image and Signal Processing*, Vol. 151, 2004, pp. 194-202.



Wei-Lieh Hsu (許威烈) received the B.S. degree in Physics in 1974, the M.S. degree in Geophysics from National Central University, Taiwan, in 1976, and the Ph.D. degrees in Computer Science and Information Engineering from National Central University, Taiwan, in 2004, respectively. Now he is an Associate Professor of the Department of Computer Information and Network Engineering at Lunghwa University of Science and Technology, Taoyuan, Taiwan. His research interests include image processing, pattern recognition, and computer networks.



Hsiao-Rong Tyan (田筱榮) received the B.S. degree in Electronic Engineering from Chung Yuan Christian University, Chungli, Taiwan, in 1984, and the M.S. and Ph.D. degrees in Computer Science from Northwestern University, Evanston, IL, in 1987 and 1992, respectively. She is an Associate Professor of the Department of Information and Computer Engineering, Chung Yuan Christian University, Chungli, Taiwan, where she currently conducts research in the areas of computer networks, computer security, and intelligent systems.



Yu-Ming Liang (梁祐銘) received the B.S. and M.S. degrees in Information and Computer Education from National Taiwan Normal University in 1999 and 2002, respectively. He is currently a research assistant with the Institute of Information Science, Academia Sinica, Taipei, Taiwan. His areas of research interest include pattern recognition, image processing, and computer vision.



Bor-Shenn Jeng (鄭伯順) received the B.S. degree from the Department of Physics, National Taiwan Normal University, Taipei, Taiwan, in 1969 and the M.S. degree from the Department of Geophysics and the Ph.D. degree from the Institute of Optical Sciences, National Central University, Taiwan, in 1973 and 1991, respectively. He has been with the Telecommunication Laboratories (TL) in Taoyuan, Taiwan, since 1974. Currently, he is the vice president of the Chunghwa Telecommunication Laboratories, where he has been working on multimedia, intelligent human/machine interface, Chinese character recognition, generation and compression, signal processing and intelligent transportation systems, etc. He has published over 100 technical papers and has over 100 patents in the area of image processing, intelligent transportation systems and pattern recognition, etc. His significant achievements include: leadership in design and deployment of the first government certificate authority in Taiwan, contribution to design and deployment of the text-to-speech system for the blind in Taiwan, leadership in design of the first ADSL-based multimedia on demand trial system in Taiwan. Dr. Jeng received the Distinguished Performance in Information Sciences Award, conferred by the Executive Yuan of Taiwan, R.O.C., in 1988 and the Distinguished Performance in Technology Award, conferred by the Executive Yuan, in 1989, which is the highest technology award conferred by the Government of Taiwan.



Kuo-Chin Fan (范國清) serves at National Central University starting from 1989 and currently is the professor of Department of Computer Science and Information Engineering and the director of Communication Research Center. He graduated from University of Florida with Ph.D. degree in 1989. He is the chairman of Image Processing and Pattern Recognition Society of Taiwan. His research is focus on pattern recognition, image analysis, and computer vision.